REMARKS

Claims 1-94 were presented for examination. The Examiner objected to the oath or declaration as failing to comply with 37 CFR 1.63(c) and 1.68. The Examiner rejected claims 1-76 and 79-94 under 35 U.S.C. §103(a) as unpatentable over Wong et al. (United States Patent Application 2003/0067485, hereafter "Wong"), in view of Dietz et al. (United States Patent No. 6,754,676, hereafter "Dietz"), and further in view of Lewallen (United States Patent No. 6,941,520, hereafter "Lewallen"). The Examiner rejected claims 77-78 under 35 U.S.C. §103(a) as unpatentable over Wong in view of Lewallen, in view of Dietz, and further in view of Snyder (United States Patent No. 6,707,475, hereafter "Snyder").

Claims 1-15, 17-24, 27, 29, 35-36, 38, 40-42, 45-48, 52, 70-73, 75, 81-84, 87, 89, and 91-94 have been amended. Claims 95-108 have been added. Claims 49, 51, 53-69, 74, 88 and 90 have been canceled. No new matter has been added. Claims 1, 21, 38, and 91 are independent.

Objection of the Oath or Declaration under 37 CFR 1.63(c) and 1.68

The Examiner stated that the declaration did not identify the mailing address of each inventor. Applicants submit herewith a supplemental Application Data Sheet providing the mailing address of each inventor and satisfying the requirements of 37 CFR 1.63 and 1.76.

The Examiner further stated that the declaration was not executed in accordance with either 37 CFR 1.66 or 1.68. Applicants enclose a copy of the executed declarations as filed on February 2, 2004, with the Response to the Notice to File Missing Parts, and a copy of the postcard submitted with the Response and stamped as received by the Office on February 2, 2004. Page 3 warns the declarants that willful false statements and the like are punishable by fine or imprisonment, or both (18 U.S.C. 1001) and may jeopardize the validity of the application or any patent issuing thereon. The declarants set forth on page 3 of the declaration that all statements made of the declarants' own knowledge are true and that all statements made on information and belief are believed to be true. Therefore, Applicants submit that the declaration satisfies the requirements of 37 CFR 1.68.

Rejection of Claims 1-76 and 79-94 Under 35 U.S.C. §103(a)

The Examiner rejected claims 1-76 and 79-94 under 35 U.S.C. §103(a) as unpatentable over Wong, in view of Dietz, and further in view of Lewallen. To establish a prima facie case of obviousness, the prior art reference (or references when combined) must teach or suggest each and every claim limitation.

The claimed invention provides a unifying data superstructure that contains the state, program code and internal logic of an application, incorporating all of the aspects of a program that would normally be controlled through private data structures, procedural code, and application programming interfaces. *See* Specification, 21. Independent claims 1, 21, 38 and 91 as hereby amended recite, in relevant part:

> a platform-independent data superstructure storing an application state, program code and internal logic of the application.

As discussed with the Examiner, Wong, Dietz and Lewallen each focus only on the generation of user interfaces to allow the use of conventionally-coded applications on diverse execution platforms. At most, a combination of Wong, Dietz and Lewallen would suggest methods for enhancing user interfaces of existing applications displayed in heterogeneous execution environments. Enhancing a user interface for display on a variety of platforms does not teach or suggest creating a superstructure representing and encapsulating a new application, including the application state, program code and internal logic of the application. One of ordinary skill in the art focused on extending the user interfaces for existing applications to a wide variety of platforms would not be motivated to provide methods and systems for creating new applications within platform-independent superstructures because the motivation in these environments is to generate new user interfaces that allow the reuse of existing applications *without* requiring the creation of new applications.

Wong fails to teach receiving a platform-independent data superstructure storing an application state, program code and internal logic of an application. A developer using the graphical user interface system described by Wong may design a platform-specific application with user interface capabilities that are independent of the different heterogeneous device platforms on which the application may run. *See* Wong, 2, paragraph 0036. However, Wong

only provides a graphical user interface system for use with a separately-developed, platform-specific application. In contrast, the claimed invention provides a platform-independent data structure that contains not only definitions for an appearance of an application but also the state, program code and internal logic of an application. *See* specification, 21.

Nor does Wong suggest a platform-independent data superstructure storing an application state, program code and internal logic of an application. The focus in Wong is on providing a system allowing existing applications to be designed with independent user interfaces, avoiding the re-design and re-implementation of applications for the graphical user interface requirements of heterogeneous platforms. *See* Wong, 1, col. 5. One of ordinary skill in the art focused on extending the user interfaces for existing applications would focus on systems allowing the reuse of existing applications and would avoid systems requiring the creation of new applications using a superstructure encapsulating the application state, program code, and internal logic of an application because the goal is to avoid the cost and labor of re-designing or re-implementing existing applications.

Dietz also fails to teach a platform-independent data superstructure storing an application state, program code and internal logic of the application. Dietz teaches a method for providing selective view of on-line surveys. *See* Dietz, col. 1, lines 34-35. The system disclosed in Dietz uses tree structures expressed in XML to represent elements of a survey presented to a user. *See* Dietz, col. 5, lines 18-20. A system for generating a survey data structure representing data collected by a survey does not teach a platform-independent data superstructure storing an application state, program code and internal logic of an application.

Nor does Dietz suggest a platform-independent data superstructure storing an application state, program code and internal logic of an application. Dietz focuses only on the use of a structure to display data to a user and does not suggest the generation of a platform-independent data superstructure defining the appearance and behavior of an application. One of ordinary skill in the art focused on preventing the loss of data collected in survey and on presenting up-to-date survey information would not be motivated to create a superstructure storing application state, program code, or internal logic of an application because the motivation is merely to

improve the storage and presentation of survey data generated by existing applications, and not on creating new applications of any kind.

Lewallen also fails to teach a platform-independent data superstructure storing an application state, program code and internal logic of an application, as recited by the pending claims. Lewallen discloses a system for generating a user interface for a program by converting a standard application programming interface (API) into a user interface API supported by a particular platform in which the application will execute – for example, by transforming the standard API into an API for accessing native operating system user interface objects and interfaces. *See* Lewallen, Abstract and at col. 5, lines 56-59. A system for converting an application programming interface of one type into an API of a second type does not teach a superstructure storing application state, program code, or internal logic of an entire application.

Nor does Lewallen suggest a platform-independent data superstructure storing an application state, program code and internal logic of an application. The focus of Lewallen is on providing Java-based user interfaces with improved look-and-feel by extending Java programs to allow access to other objects and programs available in commonly used user interface programs. *See* Lewallen, col. 6, lines 45-55, and col. 5, lines 30-40. One of ordinary skill in the art focused on improving a user interface for an existing Java application would not be motivated to provide a system for the generation of new applications through the use of a superstructure storing application state, program code, and internal logic of an application because a new method for creating an application would not provide a means for improving the user interface of an existing application.

Accordingly, Applicants submit that neither Wong nor Dietz, nor Lewallen, alone or in combination, disclose, teach or suggest each and every limitation of independent claims 1, 21, 38, and 91 or of the dependent claims. Applicants respectfully request that the Examiner reconsider and withdraw the rejection of the pending claims.

Rejection of Claims 77-78 Under 35 U.S.C. §103(a)

The Examiner rejected claims 77-78 under 35 U.S.C. §103(a) as unpatentable over Wong, in view of Dietz, further in view of Lewallen, and further in view of Snyder. To establish a prima facie case of obviousness, the prior art reference (or references when combined) must teach or suggest each and every claim limitation.

The arguments made above with respect to Wong's, Dietz's, and Lewallen's failure to teach or suggest the each and every limitation of the pending claims, apply with equal force here and are reiterated as if set forth in full. Snyder fails to teach a superstructure storing application state, program code, or internal logic of an application. Snyder describes a system for selecting and displaying navigational information to provide more efficient and intuitive methods for planning and programming navigational paths. *See* Snyder, Abstract. A system for displaying navigational information does not teach a superstructure storing application state, program code, or internal logic of an application.

Nor does Snyder suggest a superstructure storing application state, program code, or internal logic of an application. Snyder provides displays for flight management systems providing pilots with more efficient, easier to use systems for displaying FMS data and procedures that relate to the arrival and departures of aircrafts. *See* Snyder, col. 2, lines 55-60. One of ordinary skill in the art would not be motivated to modify a system for intuitive display of navigational information to generate a platform-independent data superstructure defining the appearance and behavior of an application because new methods for the creation of applications would not improve the display of navigational information for existing methods of planning navigational paths.

Accordingly Applicants submit that neither Wong nor Dietz nor Lewallen nor Snyder, alone or in combination, disclose, teach or suggest each and every limitation of claims 77-78. Applicants respectfully request that the Examiner reconsider and withdraw the rejection of claims 77-78.

CONCLUSION

In view of the above, each of the presently pending claims in this application is believed to be in immediate condition for allowance. Accordingly, the Examiner is respectfully requested to pass this application to issue.

Please charge any fees associated with this filing, or apply any credits, to our Deposit Account No. 03-1721.

Respectfully submitted,
CHOATE, HALL & STEWART LLP

Date: December 21, 2007

/John D. Lanza/
John D. Lanza
Registration No. 40,060

Patent Group
CHOATE, HALL & STEWART LLP
Two International Place
Boston, MA 02110
Tel: (617) 248-5000
Fax: (617) 248-4000